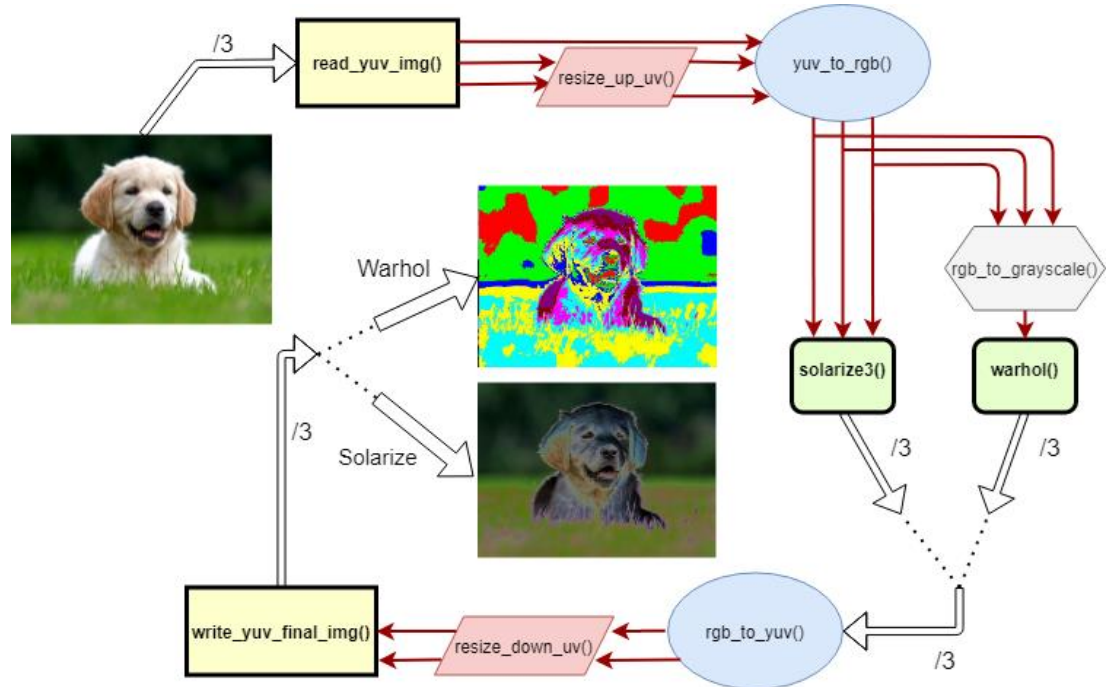


Embedded Systems Design

Δημιουργία εφέ solarization και Warhol
στην επεξεργασία εικόνας

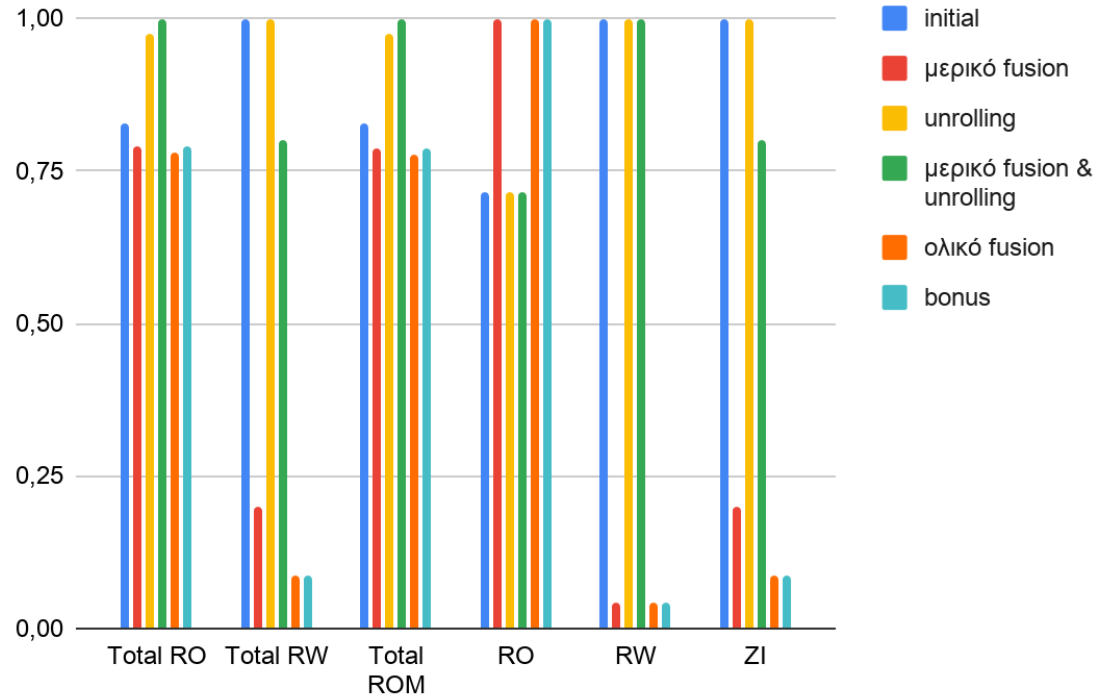
Part 1

- Αρχική υλοποίηση
- Μετασχηματισμοί για βελτίωση απόδοσης του αλγορίθμου
 - Πίνακας δεδομένων
 - Ταχύτητες εκτέλεσης



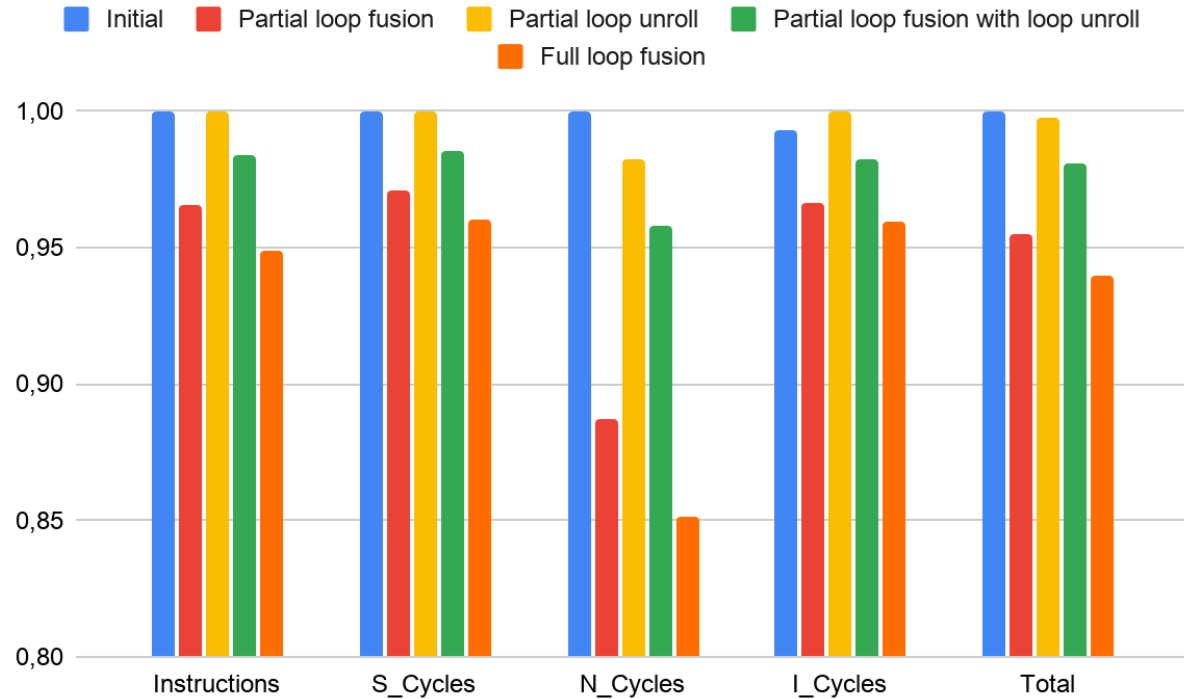
Part 1

- Αρχική υλοποίηση
- Μετασχηματισμοί για βελτίωση απόδοσης του αλγορίθμου
 - **Πίνακας δεδομένων**
 - Ταχύτητες εκτέλεσης



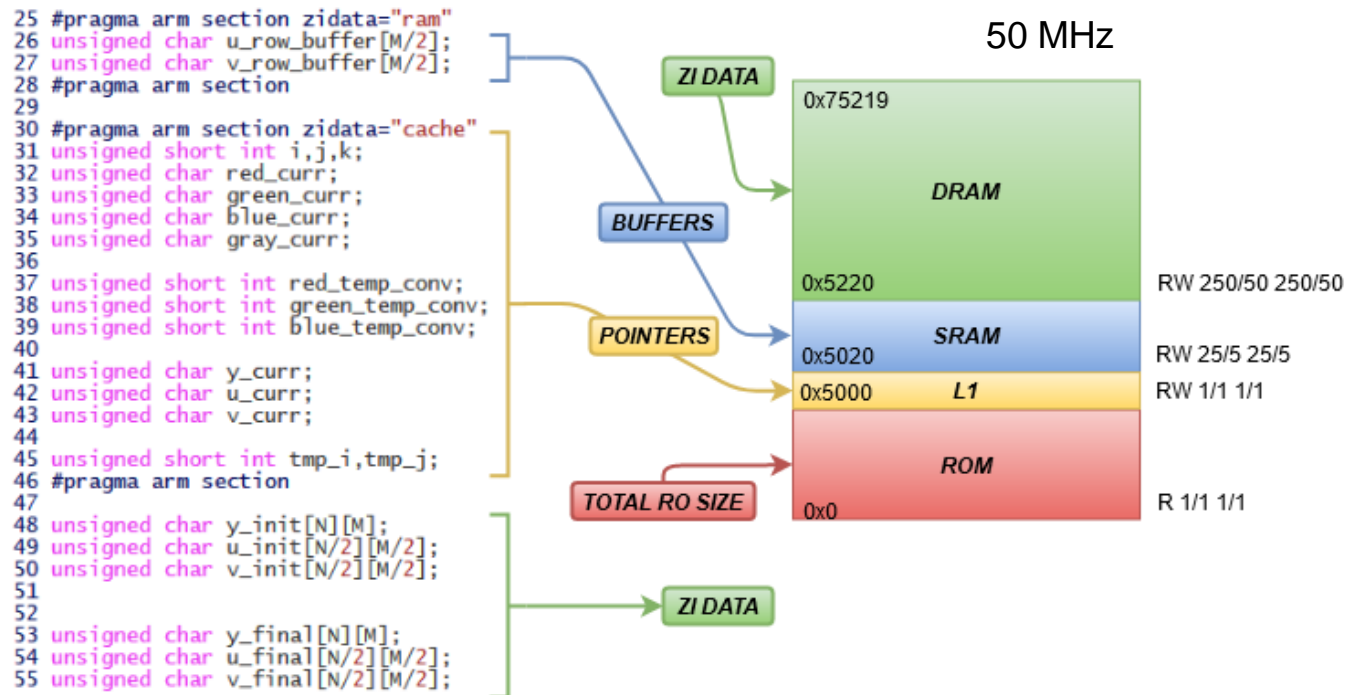
Part 1

- Αρχική υλοποίηση
- Μετασχηματισμοί για βελτίωση απόδοσης του αλγορίθμου
 - Πίνακας δεδομένων
 - **Ταχύτητες εκτέλεσης**



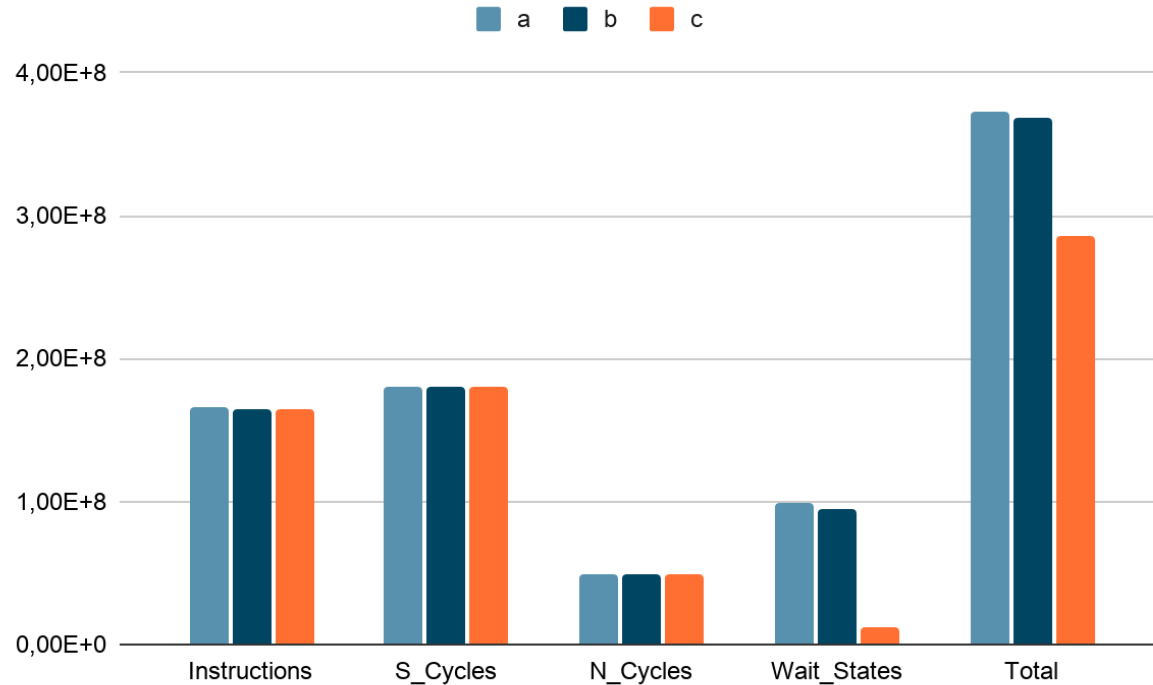
Part 2

- Ιεραρχίες μνήμης
 - **Τελική**
- Αποτελέσματα



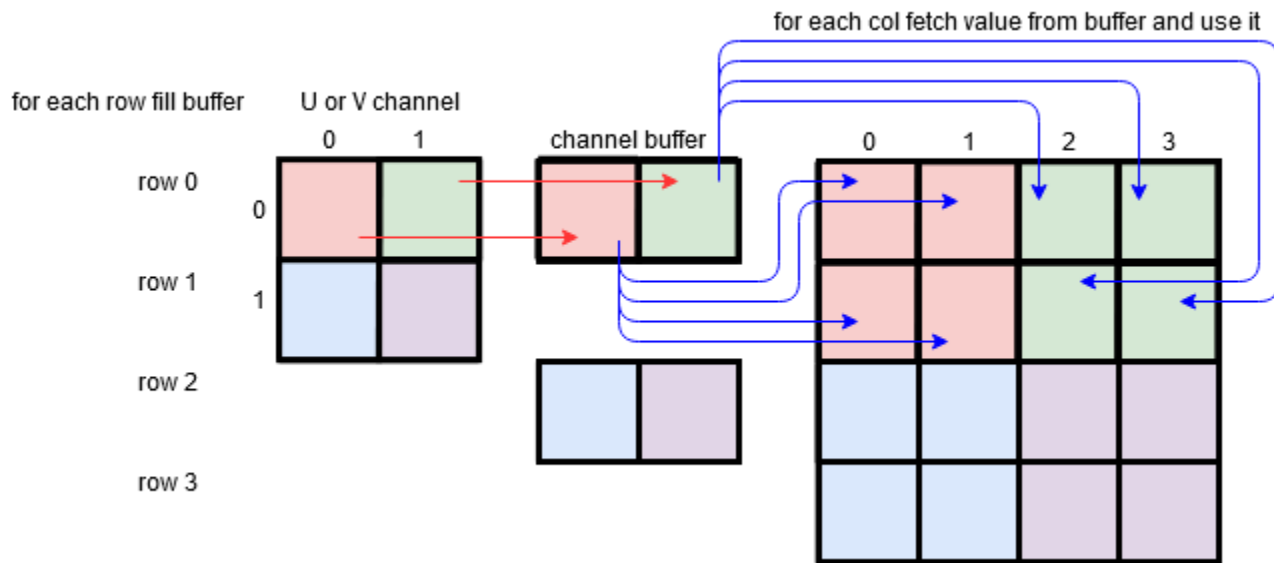
Part 2

- Ιεραρχίες μνήμης
 - Τελική
- **Αποτελέσματα για κάθε υλοποίηση**



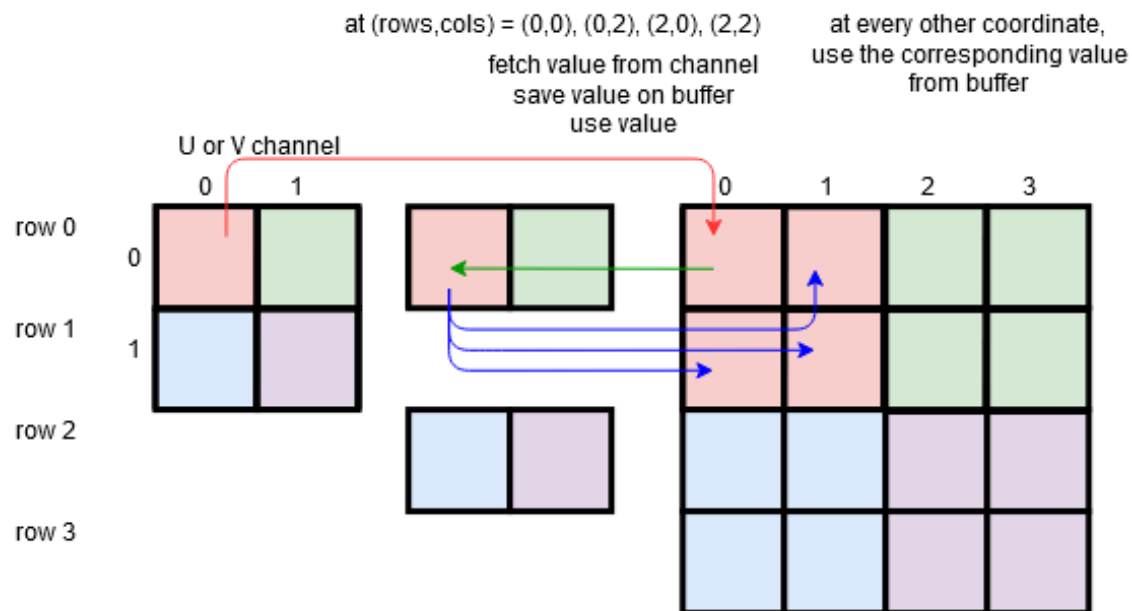
Part 3

- Επαναχρησιμοποίηση δεδομένων
 - **version 1**
 - version 2
- Αποτελέσματα για κάθε version



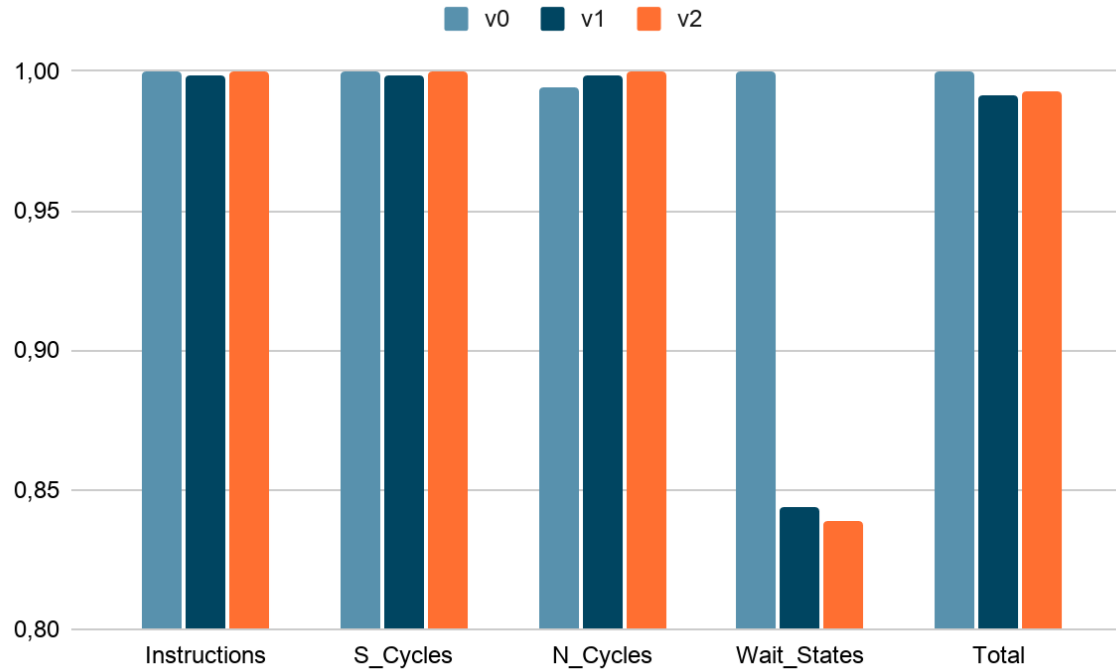
Part 3

- Επαναχρησιμοποίηση δεδομένων
 - version 1
 - **version 2**
- Αποτελέσματα για κάθε version



Part 3

- Επαναχρησιμοποίηση δεδομένων
 - version 1
 - version 2
- **Αποτελέσματα για κάθε version**



Συμπεράσματα - Καλύτερες υλοποιήσεις

- Part 1 - Μετασχηματισμός βρόγχων : **full loop fusion**

Με την εφαρμογή loop fusion στον κωδικα μας έχουμε τις καλύτερες επιδόσεις όσο αφορά τόσο τον πίνακα δεδομένων όσο και τις ταχύτητες εκτέλεσης. Εκμεταλλευόμενοι την αρχή της τοπικότητας αναφοράς πετυχαίνουμε πιο κανονικοποιημένες αναφορές στην μνήμη και συνεπώς μικρότερους χρόνους αναμονής οι οποίοι σε συνδυασμό με τον μειωμένο αριθμό εντολών οδηγούν σε αυξημένη ταχύτητα εκτέλεσης.

- Part 2 - Ιεραρχία μνήμης : **υλοποίηση c**

Αποθηκεύοντας τις συχνά χρησιμοποιούμενες μεταβλητές σε γρηγορότερες μνήμες και εφαρμόζοντας επαναχρησιμοποίηση δεδομένων κερδίσαμε γρηγορότερες και λιγότερες προσπελάσεις άρα και τελικά βέλτιστες ταχύτητες εκτέλεσης. Οπότε τα μειωμένα Wait States έπαιξαν σημαντικό ρόλο στο αποτέλεσμα.

- Part 3 - Τεχνική επαναχρησιμοποίησης δεδομένων : **version v1**

Με την τακτική που εφαρμόζεται στην **v1**, (**line buffering**) καταφέρνουμε να επανα-χρησιμοποιήσουμε δεδομένα όποτε είναι αυτό αναγκαίο, χωρίς περιττούς ελέγχους. Αποδεικνύεται ότι η σειριακή προσπέλαση έχει βαρύτητα και ότι λιγότερες προσπελάσεις δεν καταλήγουν απαραίτητα σε καλύτερους χρόνους, ιδιαίτερα όταν προστίθενται επιπλέον διακλαδώσεις, όπως στην περίπτωση της **v2**.